DEVELOPMENT OF INFERENTIAL MEASUREMENT FOR AIR DENSITY USING NEURAL NETWORK

SHANKAR RAMAKISHAN

A report submitted in fulfilment of the
requirements for the award of the degree of
Bachelor of Chemical Engineering

Faculty of Chemical & Natural Resources Engineering
Universiti Malaysia Pahang

MAY 2008

I declare that this thesis entitled “*Development of Inferential Measurement for Air Density using Neural Network*“ is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature	:	………………………………………..
Name	:	SHANKAR RAMAKISHAN
Date	:	16 MAY 2008

To my beloved father and mother

for their love and support

ACKNOWLEDGEMENT

This thesis is the end of my long journey in obtaining my degree in Chemical Engineering. After all those hard work, I've got quite a list of people who contributed in some way to this thesis, for which I would like to express thanks.

Foremost, this work would not have been possible without the support and encouragement of my supervisor, Cik Sureena Bt Abdullah, under whose supervision I chose this topic and began the thesis. With her enthusiasm, her inspiration, and her great efforts to explain things clearly and simply, she helped to make the research fun for me. Throughout my thesis-writing period, she provided encouragement, sound advice, good teaching, good company, and lots of good ideas. I would have been lost without her. It was a great pleasure for me to conduct this thesis under her supervision.

Needless to say, that I am grateful to the lecturers at UMP whose friendship and support have made it more than a temporary place of study for the students. They have been supported me throughout my research. Without their support it would have been impossible for me to finish this work.

I cannot end without thanking my family, on whose constant encouragement and love I have relied throughout my time at the UMP. Their unflinching courage and conviction will always inspire me to do my best in all matters of life. To them that I dedicate this work.

ABSTRACT

In many industrial processes, the most desirable variables to control are measured infrequently off-line in a quality control laboratory. In these situations, use of advanced control or optimization techniques requires use of inferred measurements generated from correlations. For well-understood processes, the structure of the correlation as well as the choice of inputs may be known a priori. However, many industrial processes are too complex and the appropriate form of the correlation and choice of input measurements are not obvious. Here, process knowledge, operating experience, and statistical methods play an important role in development of correlations. This paper describes a systematic approach to the development of nonlinear correlations for inferential measurements using neural networks. A three-step procedure is proposed. The first step consists of data collection and preprocessing. Next, the process variables are subjected to simple statistical analyses to identify a subset of measurements to be used in the inferential scheme. The third step involves generation of the inferential scheme.

ABSTRAK

Dalam kebanyakan proses yang dijalankan di industri, pembolehubah-pembolehubah yang penting untuk dikawal adalah diukur secara “off-line” dalam makmal kawalan kualiti. Dalam situasi sebegini, pengunaan kaedah kawalan yang canggih atau teknik pengoptimuman memerlukan ukuran yang diperolehi melalui korelasi. Untuk proses yang difahami sepenuhnya, struktur korelasi dan pilihan input diketahui selepas kajian dijalankan. Sungguhpun begitu, kebanyakan proses yang dijalankan di industri adalah terlalu kompleks dan gaya sesuai korelasi dan pilihan ukuran input adalah kurang jelas. Dengan ini diketahui bahawa pengetahuan mengenai sesuatu proses, pengalaman mengoperasi dan kaedah statistik memainkan peranan penting dalam penghasilan satu sistem korelasi. Kertas kerja ini mengkaji satu pendekatan sistematik untuk menghasilkan sistem korelasi berdasarkan ukuran-ukuran inferens menggunakan Neural Network. Tiga langkah bertatacara telah dicadangkan untuk melaksanakan kajian ini. Langkah pertama adalah pengumpulan data dan pemprosesan data. Selepas itu, pembolehubah-pembolehubah proses dianalisis untuk mengenalpasti satu subset ukuran yang sesuai untuk digunakan dalam kaedah pengukuran inferens. Langkah ketiga melibatkan penghasilan kaedah pengukuran inferens.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ANN	Artificial Neural Network
BL	Boltzmann learning
BP	backpropagation
CL	competitive learning
DCS	distributed control system
ECL	error-correlation learning
ELM	Elman network
FF	feedforward network
FPM	first principle model
LM	Levenberg-Marquardt
MAPE	mean absolute percentage error
MSE	mean square error
PI	proportional integral
PID	proportional integral derivative
PLS	partial least squares
RMSE	root mean square error
SSE	sum square error

LIST OF SYMBOLS

b	internal bias
f	transfer function
N	total number of data
N_{hdn}	number of hidden nodes
N_{inp}	number of input nodes
N_{out}	number of output nodes
N_{trn}	number of training data
N_{wgh}	total number of weights
v_i	variables
v_i^{max}	maximum value of variables
v_i^{min}	minimum value of variables
v_i^{norm}	normalize variables
w_{i}	weight factor
x_{i}	neuron input
y	neuron output
λ_1	minimum value of interval
λ_2	maximum value of interval

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background of Study

Over the years, the application of Artificial Neural Network (ANN) in process industries has been growing in acceptance. This is because ANN is capable of capturing process information in a black box manner. Given sufficient input output data, ANN is able to approximate any continuous function to arbitrary accuracy. This has been proven in various fields such as pattern recognition, system identification, prediction, signal processing, fault detection and others (Demuth and Beale, 1992).

In general, the development of a good ANN model depends on several factors. The first factor is related to the data being used. This is consistent with other black box models where model qualities are strongly influenced by the quality of data used. The second factor is network architecture or model structure. Different network architecture results in different estimation performance. Commonly, multilayer perceptron and its variances are widely used in process estimation. The third factor is the model size and complexity. What is required is a parsimonious model. This is because a small network may not able to represent the real situation due to its limited capability, while a large network may overfit noise in the training data and fail to provide good generalization ability. Finally, the quality of a process model is also strongly dependent on network training. This stage is essentially an identification of model parameters that fits the given data; and is perhaps the most important factor among all (Sexton *et al*., 2002).

In this research, the ANN model is used for inferential estimation of air density in a Gas Flow Pressure Temperature Control Training System. The aim is to address the difficulty in measuring air quality in process plants. Most quality variables in process industries require some kinds of analysis to be carried out. The use of online analyzer for product quality variables has been limited due to large measurement delay, the need for frequent maintenance as well as high capital and operating costs.

In order to adapt to market conditions while maximizing profit, the demand for accurate inferential estimators for controlling the product quality variable becomes paramount. For this reason, this study introduces ANN as means of improving inferential measurement.

1.2 Problem Statement

Due to the complexity of process plants, the amount of information required to measure the variables is dependant both on the physics and the level of precision of analysis tools. Although physical experimentation provides accurate environmental measurements without the need for modeling assumptions, a comprehensive analysis would not only require expensive equipment, but would also require large amounts of time. Numerical modeling techniques as ANN can offer an effective method of measuring the air density under various design conditions within a virtual environment. Thus the amount of physical experimentation can be reduced considerably, although, as of yet, not eliminated.

1.3 Objective

The objective of this work was to develop an inferential measurement system for air density using Neural Network which is incorporated with Matlab.

1.4 Scope of the Research

To fulfill the objective, the following scope of research was carried out:

i. Data Collection using AFPT plant available in FKKSA lab
ii. Development of ANN based inferential estimator for air density using other secondary measurements using MATLAB
iii. Evaluation of modelling using experimental results of particular AFPT plant

1.5 Contribution of the Research

As over many years chemical plant systems have become tailored for specialized production, the need to maintain control over the primary output so that the requirements of the production system can be continuously met has become even more important. So, it is important to understand the interaction of all variables, alongside their contribution to the product quality. For this reason, this work has proposed an enhancement to inferential measurement. The ANN model was implemented in inferential estimation and control scheme for air density.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The birth of Artificial Neural Network (ANN) was believed to be founded by the fact that brain is far superior compared to conventional computation techniques. Although conventional computation techniques may perform better in task requiring high degree of numerical computation and repeatable steps, our powerful brain is fault tolerant and is able to perform parallel computation. The brain is also adaptive to new environment and is capable of interpreting imprecise information. Due to this reason, scientists have been trying to apply the knowledge gained in neural biology in the effort to improve the performance of conventional computing (Bhartiya *et al.*, 2000).

Over the past few decades, ANN has generated considerable interest among researchers and various different courses of ANN research have been explored. These included network architecture and training algorithm. As a result, different types of ANN model were developed. These models were implemented in diverse field including computer science, medicine, mathematics, physics, and engineering. The amount of research activities are expected to grow to improve the performance of ANN in various applications. This would be facilitated by the advancement in computing technology that enables complex network to be implemented. It is not a surprise that the ultimate ANN model would be a powerful, robust and reliable tool to be implemented in various areas (Demuth and Beale, 1992).

2.2 Overview of Artificial Neural Network

Artificial Neural Network (ANN) is collections of mathematical models that emulate the real neural structure of the brain. In general, ANN is made up of individual interconnected simple processing elements called neurons, arranged in a layered structure to form a network that capable of performing massively parallel computation. Architecture of a general ANN is illustrated in Figure 2.1.

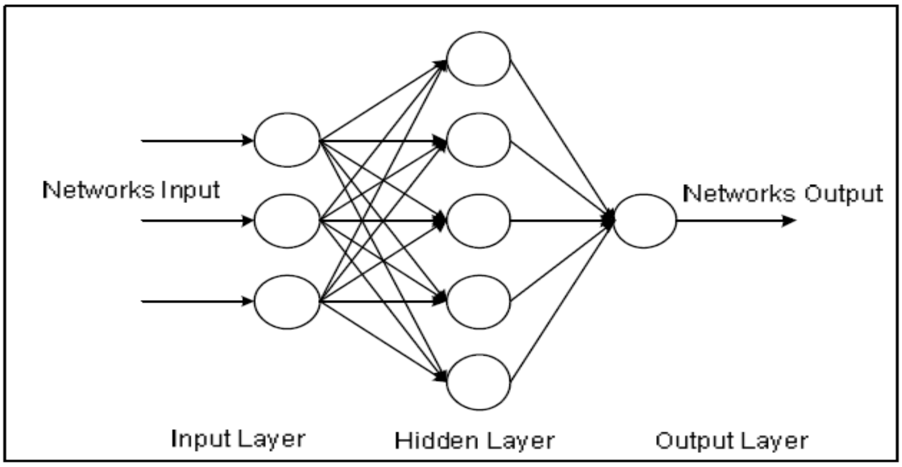

Figure 2.1: Architecture of Artificial Neural Network (Basheer & Hajmeer, 2000)

ANN can perform a human-like reasoning, learns and stores the relationship of the processes on the basis of the available representative data set. By mimicking the network of real neuron in the human brain, ANN performs mapping from an input space to an output space. Generally, the ANN does not need much of a detailed description or formulation of the underlying process. Depending on the structure of the network, a series of connecting neuron which weights are adjusted in order to fit a series of inputs to another series of known outputs. Since the connecting weights are not related to physical identities, the approach is considered as a black-box model. Such methods provide an analytical alternative to conventional techniques which are often limited by strict assumptions of normality, linearity, variable independence and so on (Hassoun, 1995).

2.2.1 Basic Element of ANN

A multilayer ANN is made up of at least three layers of neurons that are connected to each other. Input layer and output layer serve to receive the information from external resources and send the results out to external receptor. That also means most of the computing process is carried out in the hidden layer. In most networks, the output layer also performs similar transformation carried out by the hidden layer (Hassoun, 1995).

An example of artificial neuron is illustrated in Figure 2.2. The neuron input, x_i, is multiplied by the corresponding weight factor, w_i, before being sent to the neuron. This is followed by performing summation of all input in the neuron body. An internal bias, b is also introduced to enhance performance of the network. The result is passed through a nonlinear activation transfer function to obtain the output y:

$$y = f\left(\sum_i^n w_i x_i + b\right) \tag{2.1}$$

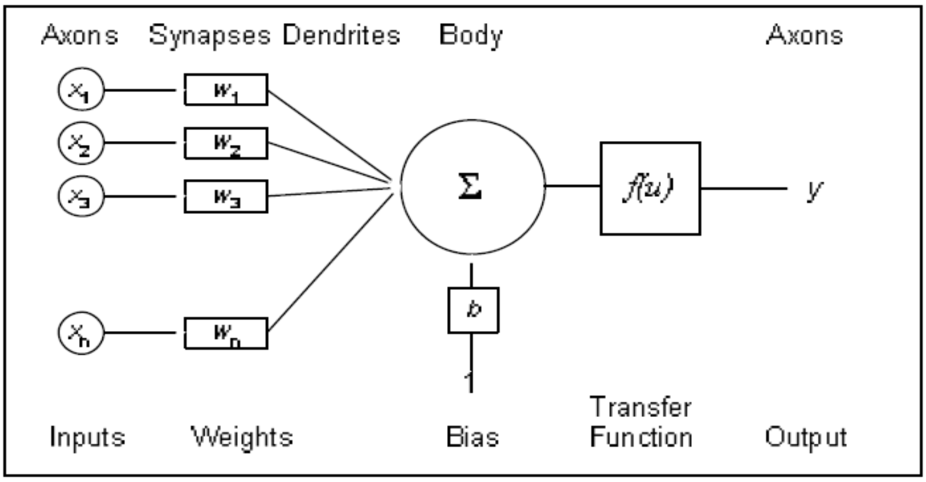

Figure 2.2: An example of artificial neuron (Hassoun, 1995)

Typical activation functions include sigmoidal function, hyperbolic tangent function, sine or cosine function. Some of these are shown in Figure 2.3. So far, there are no rules for the selection of transfer function but the sigmoidal function is the most popular choice. Besides, it is also not conclusively understood that the use of different types of transfer function will have major effect on the network performance (Demuth and Beale, 1992).

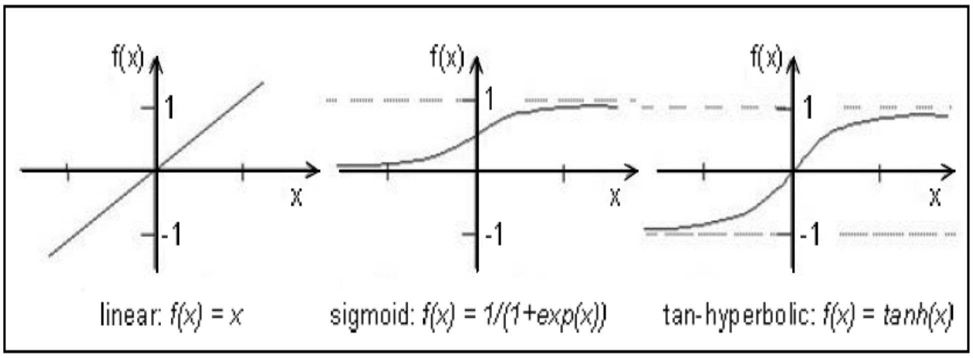

Figure 2.3: Different types of transfer function
(Demuth & Beale, 1992)

2.2.2 Network Topology

Topology of an ANN refers to how the inner structure is or how the neurons are interconnected. Generally, each neuron's output from previous layer feeds into all neurons in the subsequent layer. In the ANN model development, the topology has to be pre-specified but leave the numerical values of weight and bias up to the training phase. The inner connection is therefore particularly important for obtaining a good result.

The various structure of an ANN can be classified into three groups by the arrangement of neurons and the connection patterns of the layers. These are the feedforward network (e.g. multilayer feedforward, radial basis), recurrent network (e.g. Elman, Hopfield) and self-organizing network (e.g. Kohonen). Different types of networks may be used for different purposes. In chemical engineering application, the most influential and mostly adopted by researchers is multilayer feedforward network. Recently, the trend of using recurrent network (Elman) is also increasing. The topology for these two types network is shown in Figure 2.4 (Demuth and Beale, 1992).

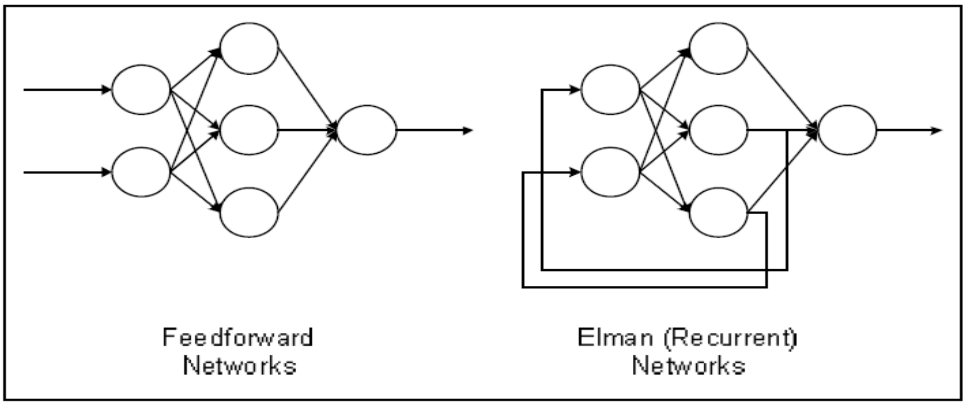

Figure 2.4: Topology of feedforward and Elman network (Demuth and Beale, 1992)

The feedforward network is named as such because the input data is transferred from input layer through hidden layers to output layers in a single direction. For Elman network, the connections are mainly feedforward but also include a set of carefully chosen feedback connections that let the network remember recent past values. The input layer is divided into two parts, i.e., the true input units and the context units that hold a copy of the activations of the hidden units from the previous time step. From the performance point of view, a feedforward network is much easier to construct and train compared to Elman network. However, since recurrent network is dynamic network that has the ability to store memory and produce output dependent of previous state of the network, it would be advantageous for the use in chemical processes since the process data are often auto correlated (Demuth and Beale, 1992).

Another important issue in ANN model development is topology selection which is referred to selection of the optimum number of hidden layers and hidden neurons. It was stated in the literature that one hidden layer is sufficient to approximate any continuous function to any desired accuracy (Irie and Miyake, 1988; Cybenko, 1989). However, some researcher used two hidden layers by considering that one hidden layer may require too many hidden neurons and this will worsen the network generalization ability and increase training time (Barron, 1994). Some researcher found that network with two hidden layers may benefit in certain specific problems. For example, Masters (1994) reported that two hidden layers may suitable for learning functions with discontinuities.

Compared to the number of hidden layers, the determination of required number hidden neurons is more complicated. Until today, the systematic way of selecting this parameter is still not well established. However, a number of rules of thumb had been proposed. In general, the optimum number of hidden neurons (N_{hdn}) is related to number of training data (N_{trn}), number of input neurons (N_{inp}), number of output neurons (N_{out}) and total number of weights (N_{wgh}). These examples are summarized in the Table 2.1. It is regretted that none of these rules can be applied perfectly to all problems.

Table 2.1: Optimum number of hidden neurons suggested

Study	Optimum N_{hdn} Suggested
Hecht-Nielsen (1990)	$N_{nhn} \leq N_{inp} + 1$
Jadid and Fairbairn (1996)	$N_{nhn} = \dfrac{N_{trn}}{R + N_{inp} + N_{out}}$; R= 5-10
Lachtermacher and Fuller (1995)	$\dfrac{0.11 N_{trn}}{N_{inp} + 1} \leq N_{nhn} \leq \dfrac{0.30 N_{trn}}{N_{inp} + 1}$
Masters (1994)	$N_{nhn} \approx \left(N_{inp} \cdot N_{out}\right)^{1/2}$
Upadhaya and Eryureka (1992)	$N_{wgh} = N_{trn} \log_2 \left(N_{trn}\right)$; N_{wgh} relate to N_{nhn}

The most practically and widely used method for optimum topology selection is trial and error search method. Basically, this was done by increasing the number of neurons from small to considerable large number of hidden neurons to be trained and then cross-validated. An increase in the number of hidden neurons used will decrease the cross-validation error. However, if too many hidden neurons are been used the network will tend to overfit the trend. Nevertheless, the final number of neuron is determined based on the smallest cross-validation error. In addition, rules of thumb may also apply as a starting point for try and error searching. Another more sophisticated method for network topology optimization is the growing and pruning methods (Sietsma and Dow, 1988).

2.2.3 Network Training and Validation

To develop an accurate process model using ANN, the learning process or training and validation are among the important steps. In the training process, a set of input-output patterns is repeated to the ANN. From that, weights of all the interconnections between neurons are adjusted until the specified input yields the desired output. Through these activities, the ANN learns the correct input-output response behaviour. For validation, the ANN is subjected to input patterns unseen during training, and introduces adjustment to make the system more reliable and robust. It is also used to determine the stopping point before overfitting occurs. A typical fitting criterion may be introduced to emphasis the model validity. Such criterion may be mean square error (MSE), sum square error (SSE) which is calculated between the target and the network output.

There are many different approaches to train the ANN. Basically, a successful learning process involves three main aspects, i.e., learning paradigm, learning rule and learning theory. Learning paradigm concerns about what information is fed to ANN. There are two types of learning paradigm, namely, supervised and unsupervised learning. In supervised learning, network is trained with the correct answer for every input data while correct answer is not provided in the unsupervised learning. Typically, most of the networks are using supervised learning except ANN model implemented in clustering or categorization.

Learning rule defines how network weight should be adjusted in the learning procedures. There are four basic types of learning rule: error-correlation learning (ECL), Boltzmann learning (BL), Hebbian learning (HL) and competitive learning (CL). Due to space limitation, the detail descriptions of these learning rules are referred to the work of Jain *et al*. (1996). Among all the training algorithms, backpropagation (BP) which follows error-correlation learning rule is the most popular choice. The famous BP is essentially a gradient steepest descent method, searching at error surface. Basically, BP involved two steps in each iteration: forward calculation to produce a solution and based on the error, backward propagation to adjust weights. However, the standard BP however is reported to suffer from several weaknesses such as slow convergence, lack of robustness and inefficiency (Rumelhart *et al*., 1986). A number of modifications have been proposed for the BP algorithm such as adaptive method and second order method to achieve better training process. Among them, Levenberg-Marquardt (LM) method which is hybrid of the Gauss-Newton nonlinear regression method and gradient steepest descent method is recommended in most optimization packages such as MATLAB.

The learning theory addresses the training data which is including issue related to data quality, data quantity and computation time. The selection data for training is important since it can affect the adaptability, reliability and robustness of an ANN. Normally, data that covers a wide range with sufficient excitation but free from outliers is preferred. Sometimes, the random noise may be injected to the training data to enhance the ANN robustness against measurement error. There is no defined rule to determine the amount of training data for ANN modelling. Generally, the data quantity is related to network structure, training method and complexity of the problem. Since ANN is needed to generalize unseen data, normally sufficient large quantity of data is needed to cover the possible unknown variable in the problem domain. The larger training data can increase the accuracy of network generalization. However, this also will increase the computation time for the learning process. Hence, there should be trade-off between these two criteria (Branke, 1995).

Another important issue regarding learning process is data normalization. The scaling of training data is needed to prevent data with larger magnitude from overriding the smaller and impede the premature learning process. Again, in this case there is no any standard approach to perform the data normalization. The simplest way is scale the variables (v_i) in the defined interval [λ_1, λ_2] using the maximum ($v_i^{\max}$) and minimum value ($v_i^{\min}$) of v_i in the database:

$$v_{norm} = \left(\frac{v_i - v_i^{min}}{v_i^{max} - v_i^{min}}\right)(\lambda_2 - \lambda_1) + \lambda_1 \qquad (2.2)$$

where v_i^{norm} is the normalize value of v_i

Training of an ANN is an optimization problem where convergence to global minimum is desired. Similar to other optimization tasks, the choice of algorithm will influence the end results. Gradient-based methods such as backpropagation provide fast convergence but are susceptible to sub-optimal solutions. On the contrary, random methods offer better probability for convergence at global minimum but can be relatively time consuming. When computing time is within acceptable level, global minimum convergence should be given consideration in this trade-off (Branke, 1995)

2.2.4 Application of ANN in Chemical Engineering

ANN is attractive due to its information processing characteristic such as nonlinearity, high parallelism, fault tolerance as well as capability to generalize and handle imprecise information (Basheer and Hajmeer, 2000). These characteristics have made ANN suitable for solving a variety of problems. The application of ANN in chemical engineering began in the late 1980's. One of the pioneering works was reported by Hoskins and Himmelblau (1988). In subsequent years, the number of research publications on ANN in chemical engineering was steadily increased. Most of these publications cover five major areas: process control, dynamic modelling, forecasting fault diagnosis, and optimization.

In the area of process control, ANN was applied through adaptive control or model-based control. By monitoring the on-line process data, ANN could be used to adjust controller parameter for optimal performance. Dynamic modelling using ANN was also well practising in process industries. By exploiting the relationship among the process variables, ANN model was developed as estimator and to be implemented in advance control techniques. Similar to dynamic modelling, forecasting can also contribute in process industries by using prediction based on the history data. This enabled behaviour of important process variable to be forecasted in the next sampling time, thus preventive action could be taken. ANN was also useful in fault diagnosis since it has the ability to store knowledge about the process and learn from the quantitative historical fault information. ANN could be trained based on the normal operating condition and then compared to current operational data to determine faults that might happen (Hoskins and Himmelblau, 1988). Lastly, ANN was implemented in plant optimization for optimal parameter searching to ensure process plant is always safe and productive.

2.2.5 Process Estimation and Control using ANN

In recent years, ANN had been extensively studied in academia as process models and controllers (Hunt *et al.*, 1992; Ungar *et al.*, 1996; Hussain, 1999; Bhartiya and Whiteley, 2001; Ahmad *et al.*, 2001). Bhat and McAvoy (1990) applied ANN to dynamic modelling for a pH-controlled CSTR. The predicted pH values when compared to two other approaches demonstrated that ANN could predict more accurately than conventional method. Willis *et al.* (1992) discussed the application of ANN as both inferential estimator and predictive controller. Their results demonstrated that ANN could accurately predict the process output and significantly improved the control scheme. Pollard *et al*., (1992) utilized backpropagation trained ANN for process identification. They concluded that ANN was particularly useful when the input-output mapping was unknown since ANN was able to accurately represent nonlinear behaviour in a black box manner. All the successful implementations of ANN in process estimation and control had proved the suitability of ANN in solving chemical engineering problems.

2.2.6 Limitation of ANN

Undeniable, ANN has been well known for its effectiveness in representing nonlinear process system. However, ANN is not a solution that can solve all problems in the real world. Among the limitation of ANN, the followings should be given added emphasis:

i. Network architecture

There is a lack of fixed rule or systematic guideline for optimal ANN architecture design. Since there is no a prior knowledge about the problem complexity, the network architecture was typically set arbitrarily. The network topology was often determined by trial and error. This subjected the network to performance uncertainties since the size of network influence the network performance: too small a network cannot learn well, but too large may lead to overfitting. Thus, algorithms that can find appropriate network architecture are needed. This includes the determination of optimum number of neuron in each layer as well as number of hidden layer needed. Many networks were developed on the assumption of being fully connected. This can be implemented on a small network but it may not be feasible for more complicated network (Hintz and Spofford, 1990).

ii. Training algorithm

The best training algorithm still cannot be singled out for general neural network. Although BP algorithm has been widely used, it does not guarantee the global optimal solution. The training may result in ANN model that is only accurate in the same operating zones as in the training data set but inaccurate in others. Besides, the selection of some parameters in BP training also lacks of systematic guideline.

iii. Training data

The quality and quantity of training data is an important issue for ANN modelling. Usually, the success of ANN relies heavily on a large amount of data, but this demand more computing time for training. In order to reduce the amount of data whilst maintaining the model quality, the data used must be carefully selected to ensure that they are sufficiently ‘rich’. This demands project understanding on the process involved. Additionally, to eliminate noise and outliers, process data may require pre-processing prior to application in neural network model development.

iv. Process relationship

Being black-box method for modelling, ANN is criticized for unable to explain and analysis the relationship between inputs and outputs. This may cause difficulties in interpreting results from the network.

2.3 Process Estimation and Control

The increase of global competitiveness has pushed chemical plant operations into highly nonlinear regions near process constraints in order to meet the ever increasing product capacity and quality. For such operating condition, process control becomes more challenging. In general, two main issues aroused with respect to process control needs. First, operating in nonlinear regions, particularly near the constraints required advanced controllers. Secondly, the limitations due to measurement difficulties must be overcome. This work concentrates on the second issue. Measurement difficulties prevail due to a variety of reasons, including: lack of appropriate on-line instrumentation and reliability of on-line instruments. Process operation has to depend on laboratory assays, which means that results can be infrequent and irregular, in addition to long analysis delays. Depending on how the laboratory analyses are carried out, the reliability of the results are being questioned too. On-line sensors may be available but they may suffer from long measurement delays (e.g. gas chromatographs) or may be subjected to factors that affect the reliability of the sensor (e.g. drifts and fouling), despite the high capital and maintenance costs. As an alternative, inferential estimation has been designed for tackling this issue.

The inferential estimator, which is designed on the basis of the model, should provide an accurate and reliable estimation even when un-measurable disturbances are present. Among the estimation approaches, ANN model show its potential to deal with nonlinear process problems. Thus, this work proposed an estimator employing ANN model to be used for this research work.

2.3.1 Inferential Estimation

An inferential control model employs measurements of secondary variables to infer the effect of un-measurable disturbances on primary process outputs such as product quality. Due to the nature of chemical and process engineering systems, the states of many variables reflect the states of other variables. By exploiting these relationships, a particular variable of interest can be represented by others in a form of correlations or models. The variable of interest termed as the primary variable which is difficult to measure can therefore be estimated using values of easy to measure secondary variables. If the model is accurate, the estimation can then serve as the replacement to actual measurement for control purposes (Doyle, 1998; Parrish and Brosilow, 1988).

Inferential estimation in chemical processes has been studied extensively since mid-1970s (Jo and Bankoff, 1976; Joseph and Brosilow, 1978). It was found that this technique is very useful and important as it can be applied to process control, process monitoring, plant fault detection and data reconciliation (Soroush 1998). Joseph (1999) investigated the application inferential estimation in a distillation column and the use of intermediate tray temperature as secondary variables in a Shell challenge case study problem. Willis *et al.* (1991) had discussed an estimation procedure for feedback control of product composition from an industrial distillation column using overhead temperature. Amirthalingam *et al.* (2000) used several tray temperatures when applying their two step identification approach to a distillation column to estimate composition in distillate.

Apart from using temperature as secondary measurement, some researchers also used flow measurements together with temperature measurement to act as inputs to estimation model. Joseph and Brosilow (1978) concluded that temperature and flow measurement can adequately estimate the compostion of debutanizer column. Similiary, Tham *et al*. (1991) used “fast” measurement of column overhead vapour temperature together with reflux flow rate to provide estimations of product concentration from a high purity distillation column.

To enhance the performance in process estimation, what is the most concern is how to design a good inferential control system. This classical problem can be divided into two categories, the selection of inferential model and the selection of a control configuration. Many types of modelling techniques have been proposed in the literature such as first principle model (FPM), partial least squares (PLS), Kalman filter, ANN and hybrid model which refers to combination of more modelling techniques. Even though these modelling techniques offered adequately good estimation in inferential control task, they still lack of adaptability to cope with dynamic environment. For this reason, ANN model has been selected as modelling technique in this work.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Stages in Methodology

The objective of this work was to develop inferential measurement system for air density using NN. To achieve this target, the research methodologies were divided into several main phases. Those were data preparation, ANN model development, and finally process estimation. The steps of these phases are summarized in the flowchart as shown in Figure 3.1.

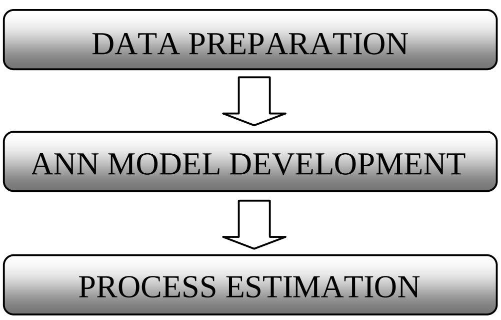

Figure 3.1: Methodology flowchart

3.2 Research Tools

Several tools have been used throughout this work. Among them, the most important software was Neural Network which is incorporated with Matlab and Model AFPT921 Gas Flow Pressure Temperature Control Training System. The training system provided the platform for all kinds of data analysis while MATLAB was used for the ANN model development. Both tools were incorporated to perform inferential measurement for air density.

3.2.1 Matlab

MATLAB is a mathematic analysis package produced by Mathworks. This program enables immediate access to high numerical computing and extended with interactive graphical capability. The entire modelling task was performed using MATLAB Version R2007A. This software provides Neural Network Toolbox for ANN model development with different types of network such as feedforward, Elman, Hopfield, Radial Basis as well as others.

3.3 Case Study – AFPT Plant

The research is carried out in local training plant, Model AFPT921 Gas Flow Pressure Temperature Control Training System which is situated in the chemical lab of Universiti Malaysia Pahang, Kuantan. The Model AFPT921 plant is a scale-down of real industrial process plant built on 5 ft x 10 ft steel platform, complete with its own dedicated control panel. The process equipment and process instrumentation are real industrial process type. The plant is constructed in accordance to industrial process plant standard and practices with fail-safe system. For example, the air heater cannot be turned on unless there is enough air flow in the pipeline. The process flowrates are at commercial production flowrates, using pipeline and not tubings.

3.3.1 Data Preparation

The research work began with the Model AFPT921 Gas Flow Pressure Temperature Control Training System for data collection. Before running the plant, the typical procedures involved were to manipulate the variables such as temperature and pressure, to define the control system and finally to start integration. Data may easily be obtained from the plant DCS (Distributed Control System) database. Data in DCS is recorded for every second as it shows the dynamic response for variables involved. The step test experiments were conducted by randomly changing selected inputs according to step size. Here three selected input variables (TIC91A, PIC91A, FIC91A) were integrated to generate sequences of random value of air density with varying temperatures and different pressure. During this process, data for all variables involved is recorded through DCS. In this work, the variables were chosen and corresponding dynamic responses in Model AFPT921 were studied. These variables are tabulated in Table 3.1.

Table 3.1: Variables in Model AFPT921

Variables	
DT	Density Transmitter
PT	Pressure Transmitter
FT	Flowrate Transmitter
TIC91A	Temperature PID Controller
TIC92A	Temperature PID Controller
TIC911A	Temperature PID Controller
FI91A	Flowrate Indicator
FIC91A	Flowrate PID Controller
PI911A	Pressure Indicator
PIC91A	Pressure PID Controller
PIC92A	Pressure PID Controller

Meanwhile the data collected is arranged and tabulated as in the format shown in table 3.2. The step change that should be manipulated is also stated in the appendix.

Table 3.2 : Tabulating the Data

Settings	**Time (min)**	**DT**	**PT**	**FT**	**Others Variables**
Pressure = 15 psig Flowrate = 20% Temperature = 40^{o}C	1 2 ..				
Pressure = 15 psig Flowrate = 20% Temperature = 60^{o}C	1 2 ..				

3.3.2 Data Conditioning

Since the performances of the resulting inferential measurement models are influenced significantly by the quality of the data used to generate them, the data collected from the process undergoes data conditioning. As the Figure 3.2 shows, the first and most important thing to do is to get rid of spurious points or outliers in the data. These can have significant impact on the model structure selection and estimator testing stages of the development cycle. Next, noise in the data should be attenuated as much as possible. On very noisy systems, this loss of predictive capabilities can be very pronounced.

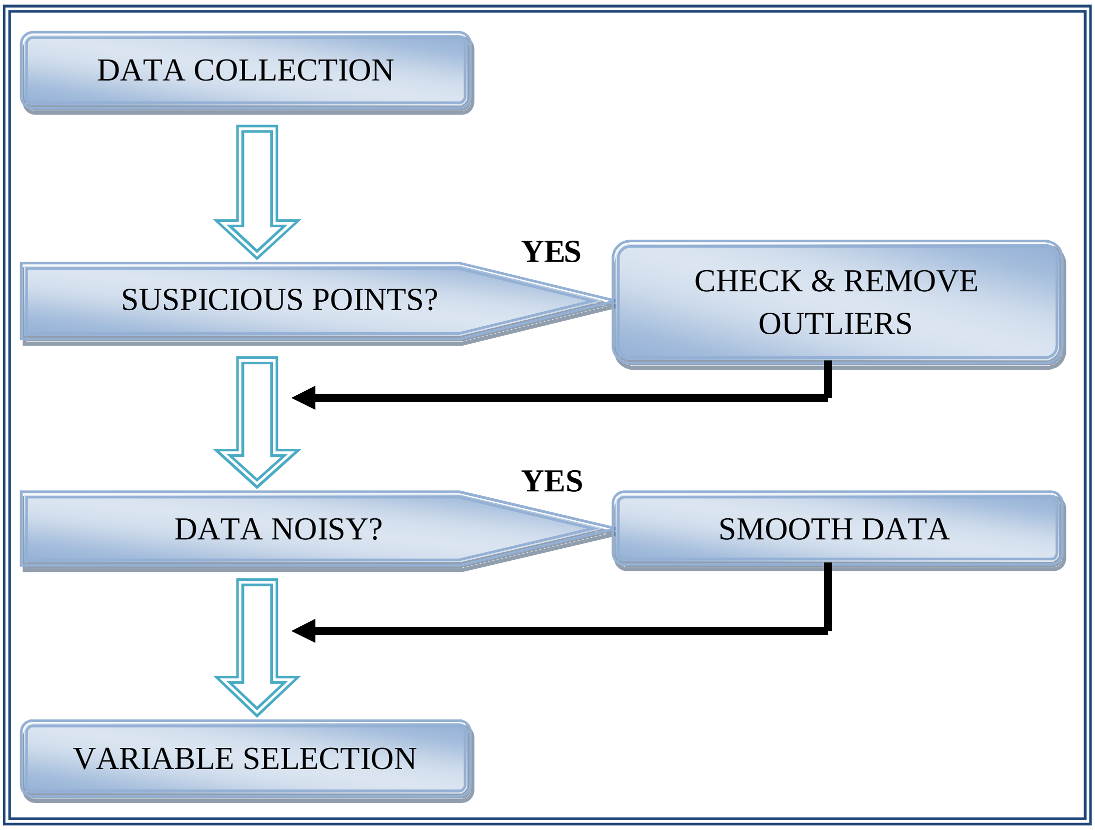

Figure 3.2 : Data Conditioning Flowchart